Look-Up Table for Superconductor Digital-RF Predistorter

Timur V. Filippov, Anubhav Sahu, Alex F. Kirichenko, and Deepnarayan Gupta

Abstract—We have developed a high-speed pipelined superconductor look-up table to generate programmable predistortion functions for direct linearization of radio frequency (RF) power amplifiers. The look-up table comprises an address decoder and a memory matrix with throughput above 10 GHz. The decoder performs code-matching of each input word and its conversion into a row address of the memory matrix. We discuss different possible implementations of the address decoder, including a preferred one for integrated circuit implementation. The memory matrix consists of RS flip-flops with nondestructive readout connected in series for slow-speed contents writing. Each row of the memory matrix contains a number, which can be read out by signal from the decoder. We present the design and the results of experimental evaluation of the look-up table and its components.

Index Terms—Decoder, memory matrix, predistortion, RSFQ.

I. INTRODUCTION

NONLINEAR high power amplifiers (HPA) create distortion that limits the dynamic range of an RF transmitter. Any effort to correct this problem decreases the amplifier's power efficiency, and at the same time increases the hardware complexity and cost. The best method to improve the transmitter's linearity is to compensate the amplifier's distortion by pre-distorting the RF waveform before it is applied to the amplifier with an inverse non-linear function. Due to speed limitation of traditional semiconductor electronics, corrective measures (such as a compensating predistortion equalizer) cannot be applied to the RF waveform directly; instead they are applied to the baseband or the intermediate frequency (IF) signals in an indirect attempt to correct the distorted RF waveform. Such baseband and IF schemes are fundamentally constrained to partial correction of weak nonlinearity over narrow bands; for large bandwidth ratios they make the situation worse [1], [2]. Therefore, a new approach is needed to linearize strongly nonlinear, but highly efficient, power amplifiers over wide frequency bands and frequency bands with large bandwidth ratios. Rapid single flux quantum technology [3], featuring ultrafast digital circuits, provides a way to generate and modify wideband RF transmit waveforms in the digital domain [4] and enables the direct RF predistortion approach.

Our first target is a predictive predistorter, where the output amplitude is a function of the input amplitude. The RF predis-
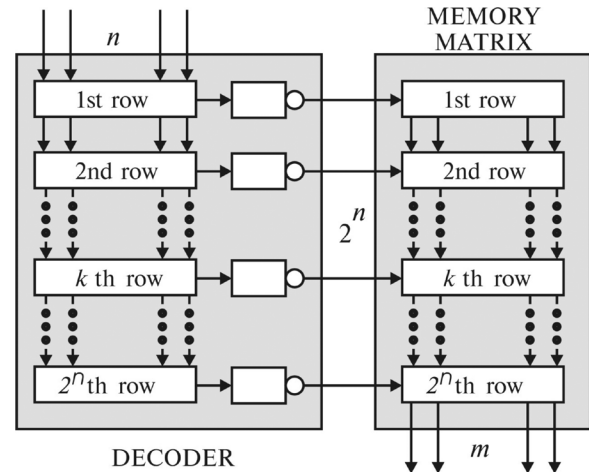
Fig. 1. The look-up table comprises n-bit decoder and a memory matrix of size $m \cdot 2^n$.

torter modifies the RF signal amplitude using a look-up table. The stored values in the look-up table determine a predistortion function that corresponds to the transfer function of a particular amplifier-chain and must be determined through calibration process.

II. LOOK-UP TABLE ARCHITECTURE

Fig. 1 shows the look-up table configuration. The input to the look-up table is the n-bit digital word, which essentially is an address of the corresponding output word. An m-bit word is stored for every possible N-bit number $(N = 2^n)$.

The decoder and the memory matrix form a pipelined structure that allows one to maintain a constant time difference between n-bit input and m-bit output words. In each clock cycle, the n-bit address decoder selects one of the stored words and reads it out in parallel through a pipelined output bus. The decoder delay complements the corresponding propagation delay of the output word, so that together the total throughput delay (latency) remains constant. If the required number is stored at address k, then the decoder requires k clock periods to decode the address and trigger read-out from the corresponding k-th row of the memory matrix (Fig. 1). Then, the additional $(2^n - k)$ clock periods are used by the memory matrix propagating the contents of k-th row to the output. The total delay of the look-up table equals to 2^n clock periods and does not depend on the address (k).

A. Address Decoder

The decoder consists of two parts,—a code matching part and a signal generating logic.

Report Documentation Page

1. REPORT DATE **JUN 2007**	2. REPORT TYPE	3. DATES COVERED **00-00-2007 to 00-00-2007**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Look-Up Table for Superconductor Digital-RF Predistorter	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **HYPRES Inc,175 Clearbrook Road,Elmsford,NY,10523**	8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT
We have developed a high-speed pipelined superconductor look-up table to generate programmable predistortion functions for direct linearization of radio frequency (RF) power amplifiers. The look-up table comprises an address decoder and a memory matrix with throughput above 10 GHz. The decoder performs code-matching of each input word and its conversion into a row address of the memory matrix. We discuss different possible implementations of the address decoder, including a preferred one for integrated circuit implementation. The memory matrix consists of RS flip-flops with nondestructive readout connected in series for slow-speed contents writing. Each row of the memory matrix contains a number, which can be read out by signal from the decoder.We present the design and the results of experimental evaluation of the look-up table and its components.

15. SUBJECT TERMS

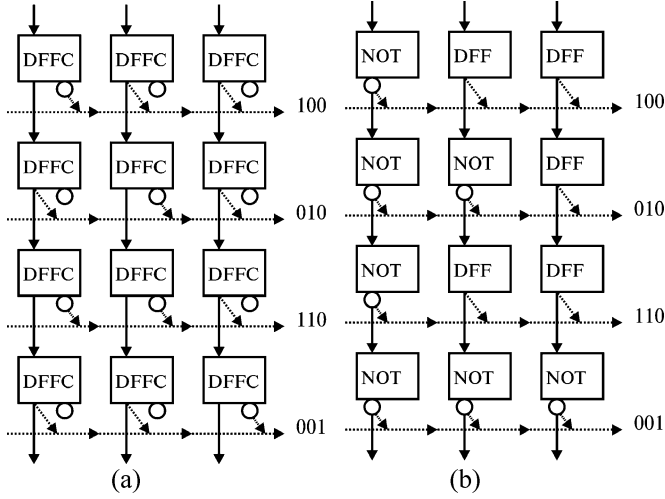| 16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**4**	

Fig. 2. Two types of a decoder code-matching matrix. (a) All cells are identical D-flip-flop with true and complementary outputs; the code value (0 or 1) for each cell is determined by the connection to either true or complementary output; data flow down the column uses only the true output of a DFFC (b) simpler cells, either D-flip-flop with true output (DFF) or with complementary output (NOT) are used. The code value for each cell is determined by the number of inversions in the column before that cell.

Fig. 3. Memory matrix consists of static memory cells, RS-flip-flops with nondestructive read out. Each row corresponds to an output word that is read out by applying the *Read* signal and merged to the output data bus using confluence buffers; D-flip-flops at each stage ensure synchronous pipelines data. Slow erasing (*Reset*) and writing (S*et*) functions are done serially.

The code-matching part of the decoder consists of D-flip-flops with complementary outputs (DFFC) [5] [Fig. 2(a)]. The basic idea of address decoding is the same as in [6]. Each row of address decoder forms a unique binary combination of ones and zeroes by connecting the corresponding true ('0') and complementary ('1') outputs of the flip-flops in that row to its output. Each input word propagates down the decoder structure, one clock period at a time. When it reaches the matching binary combination, a read-out signal (*Read*) is sent to the corresponding row of the matrix.

We considered two types of code-matching scheme and corresponding signaling logic, and decided to use the "all zeroes logic" that simplifies the decision-making logic to an n-input NOR circuit (implemented with n mergers followed by a NOT cell, in our case). If the code matching part produces at least one pulse, the corresponding signal-generating part halts the signal.

The code-matching part shown in Fig. 2(a) uses identical cells and is logically simple. In each column, the true output of each DFFC is connected to the data input of the next DFFC below it, forming a shift register. The code value is determined by hard-wiring either direct output ('0') or inverted output ('1').

Each cell in the code-matching matrix performs two functions: (1) it produces an output to the signaling logic part, and (2) it allows synchronous data-flow down the column to the cell in the next row.

Since each DFFC works either as a D-flip-flop (DFF) or as an inverter (NOT), we can simplify the circuit by choosing only one of them for each cell [Fig. 2(b)]. Logically, this scheme is more complex because one has to account for inversions in the data flow-down path (in contrast to [6]). One can do this by configuring the code-matching matrix column-by-column, by placing a NOT cell to change the value (0-to-1 and 1-to-0) and a DFF cell when no change is needed. Featuring inherent pipelining, this decoder satisfies the requirement of high-speed pipelined data flow in the entire look-up table.
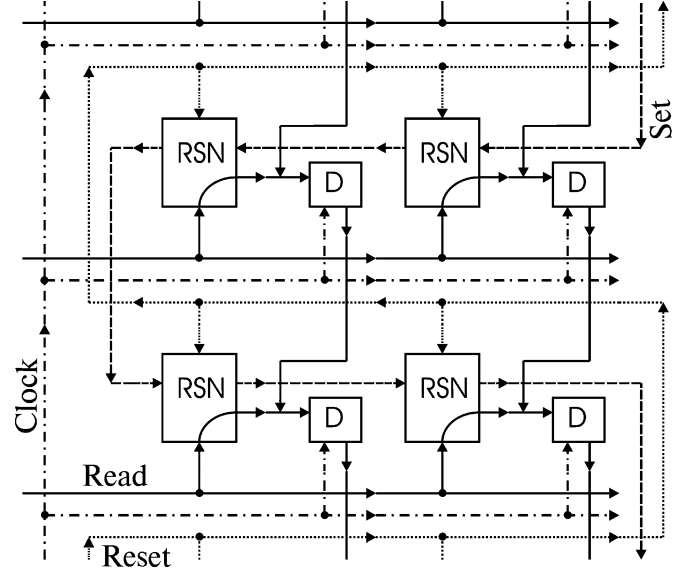
The address decoder was designed with the new code-matching scheme, which uses a combination of DFF and NOT cells. These cells, functionally complementary, were designed to have identical size and input/output/bias configuration. The output of each DFF or NOT cell is split into two channels. The first propagates to the DFF/NOT cell of the next row in the decoder, and the second proceeds to the signal-generating logic for that row.

B. Memory Matrix

The memory is constructed as a matrix of RS flip-flops with nondestructive readout (RSN). Each row contains m RSN cells (Fig. 3). When a row receives *Read* signal from the address decoder, the contents of each cell is placed on the output data bus and proceeds downwards through a chain of DFFs under clocked control. Bits cannot collide in DFF cells, because address cannot match more than one row of the decoder.

Writing and erasing the contents of each memory cell is done using *Set* and *Reset* signals. These functions, since they do not need to be fast for the present application, are done serially by connecting the Set and Reset terminals of the RSN cells to a shift register. We have designed two flavors of RSN cells with mirrored data flow—left-to-right and right-to-left in alternating rows—for optimum signal routing.

This look-up table does not need to be rewritten often like a random access memory [7], although it must be periodically updated to track any changes of the HPA characteristics. In that regard, this memory is functionally an EEPROM (Electrically Erasable and Programmable Read Only Memory). The refresh/re-calibration rate can be very slow (minutes to days). That is why we use a serial writing scheme to reduce the number of I/O wires, which would contribute to heat leak, affecting the thermal package design.
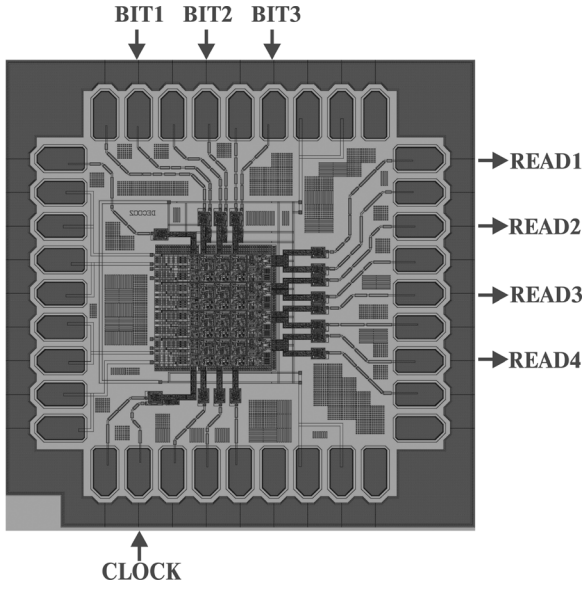
Fig. 4. Microphotograph of a 3×4 decoder low-frequency test chip. Each decoder module, containing either DFF or NOT, occupies $170 \, \mu\text{m} \times 300 \, \mu\text{m}$, and uses 24 Josephson junctions.
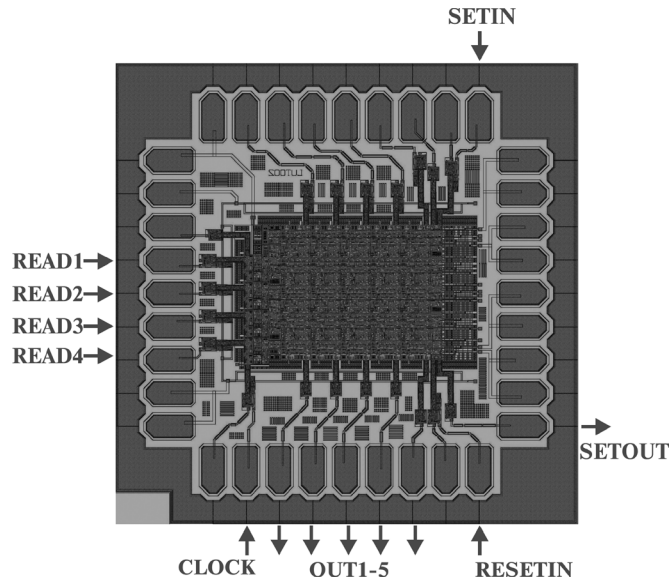

Fig. 6. Microphotograph of a look-up table low-frequency test chip comprising a 3×4 decoder and a 4×3 memory matrix.

Fig. 5. Microphotograph of a 4×5 memory matrix low-frequency test chip. Each single-bit memory module occupies $300 \, \mu\text{m} \times 325 \, \mu\text{m}$, and uses 46 Josephson junctions.

Fig. 7. Experimental waveforms from the 3×4 decoder test chip. The decoder input bits (BIT 1–3) are shown along with the decoder outputs (READ 1–4). All four possible inputs corresponding to the 4 hardwired codes are marked.

We used counterflow clock scheme for the whole look-up table. The clock pulses distribute along the left edge of the decoder and split to run along look-up table row formed by n modules of the decoder, one signaling element (inverter in our case) and m RSN modules of the memory matrix.

III. TESTING OF LOOK-UP TABLE ELEMENTS

Our testing approach is similar to the one described in [8] and is based on continual comparison of experimental data with the predictions of a computer logical simulator. The simulator includes mathematical d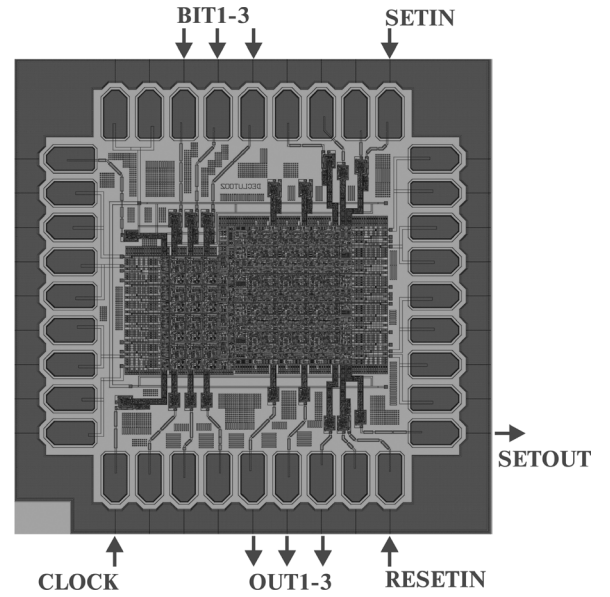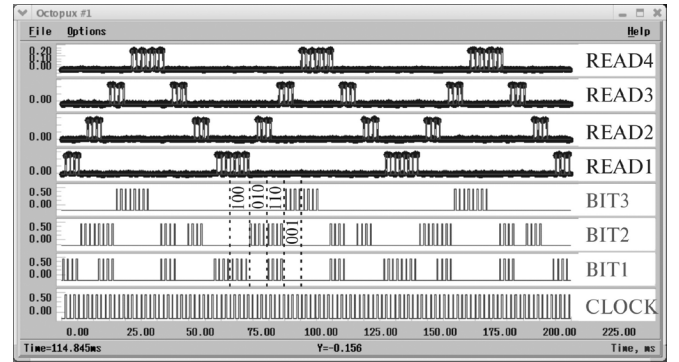escription of all cells and their responses on data and clock pulses. We were able to compare any measured response at output terminals with simulator prediction at the end of each clock period.

There were three different $5 \, \text{mm} \times 5 \, \text{mm}$ chips designed, fabricated using HYPRES $1 \, \text{kA/cm}^2$ process [9], and successfully tested: 3×4 decoder (Fig. 4), 4×5 memory matrix (Fig. 5), and look-up table comprising a 3×4 decoder and a 4×3 memory matrix (Fig. 6). We followed a comprehensive measurement procedure with the automated Octopux test system [10]. For example, the test of each bias point for a 4×5 memory matrix takes more than 400 quasi-random test vectors. The measured margins ranged from $\pm 15\%$ to $\pm 34\%$.

Fig. 7 shows results of experimental testing of a 3×4 decoder chip that is designed to implement the decoder shown in Fig. 2(b). For illustrative purposes, the decoder chip was tested row by row by applying the corresponding test vectors. According to Fig. 2(b) the test vector '100', for example, matches the first upper row only and produces the corresponding *Read* pulse. Each row was tested 4 times consecutively.
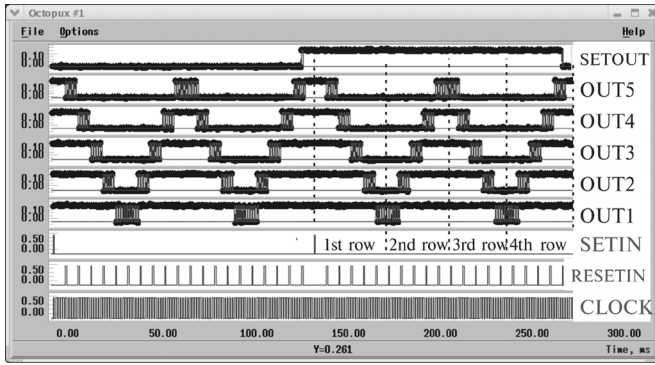
Fig. 8. Experimental waveforms from the 4×5 memory matrix test chip., showing 5 output bits (OUT 1–5). Read signals are all '0's, corresponding to the all-zeroes logic, and therefore, not shown.
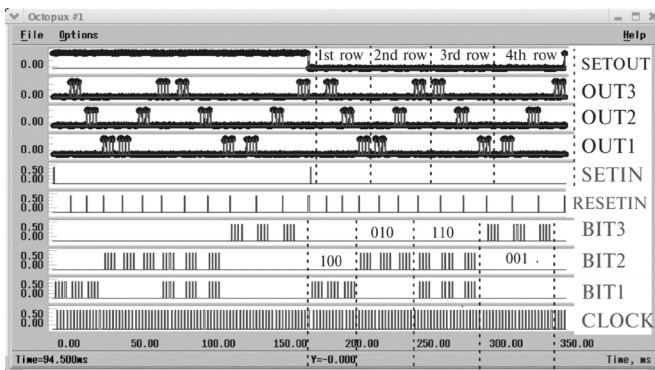

Fig. 9. Experimental waveforms from the $3 \times 4 \times 3$ look-up table test chip. A single '1', applied from the set input at the upper right corner of the memory matrix, meanders through the serially connected RSN cells through the entire memory matrix. The decoder inputs (BIT 1–3) were applied in a pattern so as to select the memory row that contains the single '1'.

Fig. 8 shows the correct operation of a 4×5 memory matrix. One single *Set* pulse is applied to the upper right RSN cell changing its state from 0 to 1. Then, $20(= 4 \times 5)$ *Reset* pulses are applied to move the nonzero state along all RSN cells connected in series.

The state of each particular RSN cell was read out 5 times by applying clock pulses and choosing the proper row of memory matrix by sending *Read* signal. The states of the upper and lower rows are read out with delay of 5 and 1 clock periods, respectively, as it is shown in Fig. 8.

The testing of a $3 \times 4 \times 3$ look-up table, comprising a 3×4 decoder and a 4×3 memory matrix, is illustrated in Fig. 9. The memory matrix was tested in the similar way to the stand-alone 4×5 matrix by moving the nonzero state along all RSN cells (4 times each cell). The row of the memory matrix was selected by applying a proper test vector to the decoder.

IV. CONCLUSION

We have developed a pipeline look-up table, the key new component of our digital-RF predistortion project. We designed, fabricated, and successfully tested stand-alone memory and decoder chips, and an integrated look-up table test chip, combining the decoder and the memory.

REFERENCES

[1] A. Katz, "Linearization: Reducing distortion in power amplifiers," *IEEE Microwave Magazine*, vol. 2, pp. 37–49, Dec. 2001.

[2] F. H. Raab, P. Asbeck, S. Cripps, P. B. Kenington, Z. B. Popovich, N. Pothecary, J. F. Sevic, and N. O. Sokal, "Power amplifiers and transmitters for RF and microwave," *IEEE Trans. Microwave Theory and Techniques*, vol. 50, pp. 814–826, Mar. 2002.

[3] K. Likharev and V. Semenov, "RSFQ logic/memory family: A new Josephson junction technology for sub-teraherz clock frequency digital systems," *IEEE Trans. Appl .Supercond.*, vol. 1, pp. 3–28, Mar. 1991.

[4] O. Mukhanov, D. Gupta, A. Kadin, J. Rosa, V. Semenov, and T. Filippov, "Superconductive digital-RF transceiver components," in *Proc. of the SDR Technical Conference*, San Diego, 2002, vol. 1, pp. 227–232 [Online]. Available: http://www.hypres.com

[5] A. F. Kirichenko, V. K. Semenov, Y. K. Kwong, and V. Nandakumar, "4-bit rapid single-flux-quantum decoder," *IEEE Trans. Appl. Supercond.*, vol. 5, pp. 2857–2860, June 1995.

[6] P. Bunyk, A. Y. Kidiyarova-Shevchenko, and P. Litskevich, "RSFQ microprocessor: New design approach," *IEEE Trans. Appl. Supercond.*, vol. 7, pp. 2697–2704, June 1997.

[7] A. F. Kirichenko, O. A. Mukhanov, and D. K. Brock, "A single flux quantum cryogenic random access memory," in *Extended Abstract of 7th International Superconductive Electronics Conference*, Berkley, 1999, pp. 124–127.

[8] T. V. Filippov, S. V. Pflyuk, V. K. Semenov, and E. B. Wikborg, "Encoders and decimation filters for superconductor oversampling ADCs," *IEEE Trans. Appl. Supercond.*, vol. 11, pp. 545–549, Mar. 2001.

[9] HYPRES Design Rules Available, HYPRES, Inc, 175 Clearbrook Rd., Elmsford, NY 10523 [Online]. Available: http://www.hypres.com

[10] D. Y. Zinoviev and Y. A. Polyakov, "Octopux: An advanced automated setup for testing superconductor circuits," *IEEE Trans. Appl. Supercond.*, vol. 7, pp. 3240–3243, June 1997.